

S/N 09/928,028

Attorney Docket No. SYN006-04

REMARKS

Claims 1-57 are pending in this application and have been rejected by the Examiner.

1. Specification

Applicant has amended the abstract of the invention in accordance with the Examiner's comments. The abstract is now between 50 and 150 words.

2. 102(e) Rejection over "Core JINI", second edition by W. Keith Edwards

The Examiner has rejected all of the claims in the application as being disclosed by Edwards, Keith, W., Core Jini, 2nd Edition (2001 Prentice Hall PTR) (referred to herein as "Edwards.") Applicant respectfully disagrees with the Examiner's position. Much of what is disclosed in Edwards is well known within the distributed computing community and much of this is recounted in the background section of the present application. This invention is a tremendous improvement on Edwards and the known art of distributed computing. Specifically, Edwards does not describe the use of a plurality of services formed into a group which act to the rest of the distributed community as a single service. Edwards does not describe a proxy for controlling such a group ("group proxy"), or a method for creating, modifying and registering group proxies.

a. Office Action - Paragraph 5 - Rejection of Claim 1

The sections of Edwards cited by the Examiner do not correspond to the elements of this claim. It is possible that the Examiner misunderstood the use of the term "group" as used by Edwards and in this application. Edwards does describe groups of Jini services that work together. However these are not groups that form a single logical unit that act as a single service

S/N 09/928,028

Attorney Docket No. SYN006-04

to the rest of the community. The type of group used in the present invention is described in the Background section of the application at page 5, lines 7-14. This is shown in Figure 2 as either the Sensor Group 50 or the Credit Group 52. With reference to Figure 1, Edwards would describe all of the clients 2, 4 and Services 10, 12, 14, 16 as a "group" because they work together.

Applicant does not claim to have invented groups and indeed describes the known art of groups at page 5, lines 7 through page 6, line 11 (describing both coordinator cohort and peer groups). Applicant's invention is directed to an improved method of forming and working with groups. Key to this improved method is the "group proxy" which, in the preferred embodiment, is a dynamically generated proxy that is capable of communicating with any of the services in a group on behalf of a client. From the perspective of the client, this makes the group look like a single service. In the prior art, in order for a client to communicate with a group the client needed to be aware of the group membership and dynamics. In the present invention the group proxy handles all of this group complexity and from the perspective of a client, the group is a single service. With the group proxy, the client calling a group service does not even need to know that the service it is calling is comprised of a group.

Grouping identical or functionally similar services in this way is crucial for building a fault-tolerant service – one that is continuously available despite the failure of some of the individual components of the group – and requires special logic to ensure the consistent externally-visible behavior of this service grouping. That logic is contained in this invention's group logic shell, and group proxy. Edwards' notion of a group is a set of unique services that can be invoked by clients individually to accomplish a task; one client might invoke A and then

S/N 09/928,028

Attorney Docket No. SYN006-04

C and D to accomplish a task, while another client invokes C, then B and D to accomplish a different task.

The group proxy is described in great detail from page 15, line 15 through page 17, line 17 of the application. Specifically, each group proxy is comprised of a group logic shell and a one or more service proxies. The service proxies are known in the prior art and are simply the proxies described by Edwards that can be used to control each service. The group logic shell is novel and is a code module that handles the grouping logic. Thus with reference to Figure 2, a group proxy 42 for the credit group 52 would be comprised of the service proxy 16a for the credit card charge 16 and the service proxy 18a for the related log service 18 in the group. (This is a peer group, in a coordinator group the group proxy only need contain the service proxy for the group leader.)

The sections of Edwards cited by the Examiner make no reference to the use of proxies to communicate with a group, a group proxy, or a group logic shell. Applicant respectfully submits that claim 1 is allowable in its present form.

b. Office Action, Paragraphs 6 & 7 – Rejection of Claims 2 and 3

As discussed above, Edwards does not disclose a mobile group proxy. In fact, the section cited by the Examiner teaches away from the present invention. Edwards states, "Jini . . . has no idea what your company's organizational structure is. So you have to tell Jini that information yourself." (Edwards, page 6, lines 5-8). As described above, and throughout the application, the client using a group proxy does not even need to know it's calling a group and it does not need to know the group's structure. Further, the cited section of Edwards does not discuss creating or

S/N 09/928,028

Attorney Docket No. SYN006-04

modifying a group proxy, or creating or modifying any proxy, while an application is executing.

Therefore, the present invention is a great improvement upon Edwards.

Applicant respectfully submits that claim 2 and 3 are allowable in their present form.

c. Office Action, Paragraph 8 – Rejection of Claims 4, 11, 18, 33, 42, 47, 52 & 56

Examiner has cited Edwards, page 65, lines 19-35 as disclosing command buffering. Buffering is storing of commands from a client to a group of services while the group is unavailable and then forwarding those commands to the group when it becomes available. Applicant is unable to find the term buffering or the concept of buffering in the cited sections of Edwards. A review of the index of Edwards does not reveal the use of the term "buffering." Even if Edwards did describe buffering to a single service, it would not deal with buffering to a group of services since it does not deal with groups.

Applicant respectfully submits that claims 4, 11, 18, 33, 42, 47, 52 & 56 are allowable in their present form.

d. Office Action, Paragraph 9 – Rejection of Claims 5, 12 & 31

The sections of Edwards cited by the Examiner describe dynamically downloading a proxy from a lookup service to use to call a service. The "proxy" described in Edwards is the equivalent of the "service proxy" in the present application. (The term "service" was added in the application to "proxy" to distinguish "group proxy" from "service proxy".) Applicant acknowledges that downloading a service proxy is in the prior art and it is described in the background section of the application at page 3, lines 17 through page 4, line 5. A point of novelty in the current location is to provide a proxy that can call a group of services instead of

S/N 09/928,028

Attorney Docket No. SYN006-04

just a single service. Edwards does not describe a group logic shell, a group proxy, or combining a service proxy with a group logic shell to form a group proxy. Edwards does not even describe working with groups.

Applicant respectfully submits that claims 5, 12, and 31 are allowable in their present form.

e. Office Action, Paragraphs 10 and 11 – Rejection of Claims 6, 7, 13, 14, 29, 35, 44 & 49

The sections of Edwards cited by the Examiner do not discuss groups, group structure (coordinator cohort or peer-to-peer) or selecting a group logic shell based upon the type of group structure. Applicant respectfully submits that claims 6, 7, 13, 14, 29, 35, 44 & 49 are allowable in their present form.

f. Office Action, Paragraph 12 – Rejection of Claims 19 & 30

Reference is made to the arguments set forth above with respect to Office Action Paragraph 5 concerning the difference between the term "group" as used by Edwards and the term "group" as used in the current application. However, even if the use of the term "groups" were the same, Edwards does not describe a system that uses a proxy to call an entire group. The proxies in Edwards are still used to call single services within the community that Edwards describes as a group. Edwards does to describe a group proxy, a group service or a group service capable of updating a group proxy.

Applicant respectfully submits that claims 19 and 30 are allowable in their present form.

S/N 09/928,028

Attorney Docket No. SYN006-04

g. Office Action, Paragraph 20 – Rejection of Claims 34, 25, 26 & 53

Reference is made to the arguments set forth above with respect to Office Action Paragraph 5 concerning the difference between the term "group" as used by Edwards and the term "group" as used in the current application. However, even if the use of "groups" were the same, Edwards does not describe a system that uses a proxy to call an entire group. The proxies in Edwards are still used to call single services within the community that Edwards describes as a group. Edwards does to describe a group proxy, a group service or a group service capable of updating a group proxy.

It may be that the Examiner believes that the lookup service described by Edwards is similar to the novel group service described in this application. This is not correct. As can be seen in Figure 2 of the application, the group service 24 is in addition to the lookup service 20. The group service 24 forms the group proxies 40,42 and registers them with the lookup service 20 so that to a client 2, 4 querying the lookup service 20, each group 50, 52 appears to act a single service.

In addition, with respect to claims 34, 25 and 26, Edwards does not describe grouping agents that attach to services and allow a plurality of services to act as a group. With respect to claim 53, Edwards does not describe buffering.

Applicant respectfully submits that claims 34, 25, 26 & 53 are allowable in their present form.

h. Office Action, Paragraph 21 – Rejection of Claims 43 and 48

S/N 09/928,028

Attorney Docket No. SYN006-04

The cited sections of Edwards do describe the equivalent of service proxy. However, they do not describe a grouping agent. Edwards does not describe, either as a separate code module or integrated into a service, an agent that handles communications with other services within a logical group of services acting together to perform a task. This is not surprising since Edwards does not discuss "groups" as used within the current application. Furthermore, Edwards does not disclose a group service or its equivalent, or a group proxy comprised of a service proxy and a group logic shell.

Applicant respectfully submits that claims 43 & 48 are allowable in their present form.

i. Remaining Dependent Claims

Applicant has already addressed each of the independent claims and many of the dependent claims above. Since all of the independent claims are allowable, as argued above, Applicant submits that the remaining dependent claims not specifically addressed herein are allowable based upon the arguments set forth with respect to both the independent claims and dependent claims specifically addressed.

Applicant respectfully submits that dependent claims 8-10, 15-17, 20-24, 27, 28, 32, 36-41, 45, 46, 50, 51, 54, 55 & 57 are allowable in their present form.

Fundamentally, the use of mobile proxies and lookup services are well known in the prior art to coordinate the interaction of independent services and clients. Groups of logical services made up of multiple services (often, but not exclusively, for redundancy purposes) but acting as a single service at the network are also well known in the prior art. The novelty of the current

S/N 09/928,028

Attorney Docket No. SYN006-04

invention is to generalize techniques used in the prior art to handle individual services and apply these techniques to groups of services, so that a client calling a group can do so with a single "group proxy" that handles the logic of group interaction. In the prior art, in order to call a group, the client was required to possess knowledge of the group, but in the current invention the client does not need to know anything about groups, group logic or group dynamics. The client merely downloads the group proxy and calls the group. From the point of view of the client, the group is a single service.

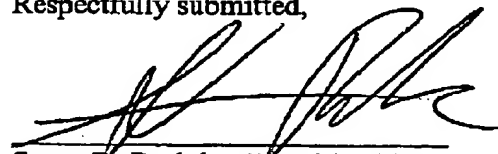
The group service described in the application handles the logic of creating, registering and modifying group proxies.

The grouping agents described in the application act on behalf of each service in a group and handle the logic and coordination of group membership on behalf of the specific service for which each acts.

It is respectfully submitted that all claims in the application are allowable. Reconsideration and withdrawal of all rejections are respectfully requested. Favorable notice to this effect and early Notice of Allowance are earnestly solicited.

Should the examiner have any questions and in order to expedite prosecution of this Application, the Examiner is encouraged to contact the undersigned directly.

Respectfully submitted,



Date: May 13, 2005

Stuart D. Rudoler, Reg. No. 45,059
Attorney for Applicant
Stuart Rudoler LLC
2 Bala Plaza, Suite 300
Bala Cynwyd, PA 19004
Tel: 610-660-7753; Fax: 267-200-0796